GAU 2757

#13

PATENT

Docket No. 1963-4728

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS:	Filepp et al.		
SERIAL NO.:	08/933,488	**GROUP ART UNIT:**	2757
FILED:	September 18, 1997	**EXAMINER:**	Moustafa M. Meky

FOR: **INTERACTIVE COMPUTER NETWORK AND METHOD OF OPERATION**

CERTIFICATE OF MAILING (37 C.F.R. 1.8a)

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Sir:

I hereby certify that the attached:

1. Reply to Office Action;
2. Amendment Fee Transmittal; and
3. Return postcard.

along with any paper(s) referred to as being attached or enclosed and this Certificate of Mailing are being deposited with the United States Postal Service on date shown below with sufficient postage as first-class mail in an envelope addressed to the: Assistant Commissioner for Patents, Washington, D.C. 20231.

Respectfully submitted,

MORGAN & FINNEGAN, L.L.P.

Dated: March 30, 2000

John Osborne
Registration No. 36,231

CORRESPONDENCE ADDRESS:
MORGAN & FINNEGAN LLP
345 Park Avenue
New York, New York 10154
(212) 758-4800
(212) 751-6849 Facsimile

PATENT

Docket No. 1963-4728

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS:	Filepp et al.	**GROUP ART UNIT:**	2757
SERIAL NO.:	08/933,488	**EXAMINER:**	Moustafa M. Meky
FILED:	September 18, 1997		

FOR: **INTERACTIVE COMPUTER NETWORK AND METHOD OF OPERATION**

AMENDMENT FEE TRANSMITTAL

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Sir:

Transmitted herewith is an Amendment for the above-identified application.

[X] No additional fee is required.

[ ] The additional fee has been calculated as shown below:

CLAIMS AS AMENDED

	Claims Remaining After Amendment		Highest No. Covered by Previous Payments	Present Extra	Rate	Additional Fee
Total* Claims	6	-	20	= 0	x $18.00	$ 0
Independent Claims	1	-	3	= 0	x $78.00	$ 0
Multiple Dependent Claim(s)	(If claims added by amendment include Multiple Dependent Claim(s) and there was not Multiple Dependent Claims(s) in application before amendment add $260.00 to additional fee.)					$ 0
					Total:	$ 0

[ ] Statement of "Small Entity" Status Under 37 CFR § 1.27 filed _____.
Reduced Fees Under 37 CFR § 1.9(f) (50% of total) paid herewith. $ _____.

* Includes all independent and single dependent claims and all claims referred to in multiple dependent claims. See 37 C.F.R. § 1.75(c).

[ ] Charge fee to Deposit Account No. 13-4500. Order No. _____.
A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

[X] The Assistant Commissioner is hereby authorized to charge any additional fees which may be required for this amendment, including all fees pursuant to 37 C.F.R. § 1.17 for its timely consideration, or credit any overpayment to Deposit Account No. 13-4500. Order No. 1963-4728.
A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

[ ] ____ Page(s) of substitute Sequence Listing

[ ] ____ Computer disk(s) containing substitute Sequence Listing

[ ] Statement under 37 C.F.R. § 1.825(b) that the computer and paper copies of the substitute Sequence Listing are the same.

[ ] A check in the amount of $ _____ to cover the filing fee is attached.

Respectfully submitted,

MORGAN & FINNEGAN, L.L.P.

Dated: March 30, 2000

By: Israel Blum
Israel Blum
Registration No. 26,710

CORRESPONDENCE ADDRESS:

MORGAN & FINNEGAN, L.L.P.
345 Park Avenue
New York, New York 10154
(212) 758-4800
(212) 751-6849 Facsimile

6. [ ] Charge fee to Deposit Account No. 13-4500. Order No. ___________. A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

7. [X] The Assistant Commissioner is hereby authorized to charge any additional fees which may be required by this paper, or credit any overpayment to Deposit Account No. 13-4500. Order No. 1963-4728. A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

Respectfully submitted,

MORGAN & FINNEGAN, L.L.P.

Dated: April 6, 1999

By: John A. Orlove /for Israel Blum
Reg. No. 36,231
Israel Blum
Registration No. 26,710

CORRESPONDENCE ADDRESS:

MORGAN & FINNEGAN, L.L.P.
345 Park Avenue
New York, New York 10154-0053
(212) 758-4800
(212) 751-6849 Facsimile

459468_1

PATENT

Docket No. 1963-4728

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS:	Filepp et al.	**GROUP ART UNIT:**	2757
SERIAL NO.:	08/933,488	**EXAMINER:**	Moustafa M. Meky
FILED:	September 18, 1997		

FOR: **INTERACTIVE COMPUTER NETWORK AND METHOD OF OPERATION**

CERTIFICATE OF MAILING (37 C.F.R. 1.8a)

COMMISSIONER OF PATENTS
Washington, D.C. 20231

Sir:

I hereby certify that the attached:

1. Formal Drawings (Figs. 1, 2, 3a, 3b, 4a, 4b, 4c, 4d, 5a, 5b, 6, 7, 8, 9, 10 and 11)
2. Transmittal of Substitute Formal Drawings; and
3. Return postcard.

along with any paper(s) referred to as being attached or enclosed and this Certificate of Mailing are being deposited with the United States Postal Service on date shown below with sufficient postage as first-class mail in an envelope addressed to the: Commissioner of Patents, Washington, D.C. 20231.

Respectfully submitted,

MORGAN & FINNEGAN, L.L.P.

Dated: June 15, 2000

John Osborne
Registration No. 36,231

CORRESPONDENCE ADDRESS:
MORGAN & FINNEGAN LLP
345 Park Avenue
New York, New York 10154
(212) 758-4800
(212) 751-6849 Facsimile

PATENT

Docket No. 1963-4728

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS:	Filepp et al.	**GROUP ART UNIT:**	2757
SERIAL NO.:	08/933,488	**EXAMINER:**	Moustafa M. Meky
FILED:	September 18, 1997		

FOR: **INTERACTIVE COMPUTER NETWORK AND METHOD OF OPERATION**

AMENDMENT FEE TRANSMITTAL

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Sir:

Transmitted herewith is an Amendment for the above-identified application.

[X] No additional fee is required.

[ ] The additional fee has been calculated as shown below:

CLAIMS AS AMENDED

	Claims Remaining After Amendment		Highest No. Covered by Previous Payments	Present Extra	Rate	Additional Fee
Total* Claims	6	-	20	= 0	x $18.00	$ 0
Independent Claims	1	-	3	= 0	x $78.00	$ 0
Multiple Dependent Claim(s)	(If claims added by amendment include Multiple Dependent Claim(s) and there was not Multiple Dependent Claims(s) in application before amendment add $260.00 to additional fee.)					$ 0
					Total:	$ 0

[ ] Statement of "Small Entity" Status Under 37 CFR § 1.27 filed _____.
Reduced Fees Under 37 CFR § 1.9(f) (50% of total) paid herewith. $ _____.

* Includes all independent and single dependent claims and all claims referred to in multiple dependent claims. See 37 C.F.R. § 1.75(c).

[ ] Charge fee to Deposit Account No. 13-4500. Order No. _____.
A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

[X] The Assistant Commissioner is hereby authorized to charge any additional fees which may be required for this amendment, including all fees pursuant to 37 C.F.R. § 1.17 for its timely consideration, or credit any overpayment to Deposit Account No. 13-4500. Order No. 1963-4728.
A DUPLICATE COPY OF THIS SHEET IS ATTACHED.

[ ] ____ Page(s) of substitute Sequence Listing

[ ] ____ Computer disk(s) containing substitute Sequence Listing

[ ] Statement under 37 C.F.R. § 1.825(b) that the computer and paper copies of the substitute Sequence Listing are the same.

[ ] A check in the amount of $ _____ to cover the filing fee is attached.

Respectfully submitted,

MORGAN & FINNEGAN, L.L.P.

Dated: March 30, 2000

By: Israel Blum
Israel Blum
Registration No. 26,710

CORRESPONDENCE ADDRESS:

MORGAN & FINNEGAN, L.L.P.
345 Park Avenue
New York, New York 10154
(212) 758-4800
(212) 751-6849 Facsimile

PATENT

ATTORNEY DOCKET NO. 1963-4728

#13 SK 4-5-00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Filepp et al.	Group Art Unit: 2757
Serial No.	:	08/933,488	Examiner: Moustafa M. Meky
Filed	:	September 18, 1997	
For	:	**INTERACTIVE COMPUTER NETWORK AND METHOD OF OPERATION**	

REPLY TO OFFICE ACTION

Assistant Commissioner for Patents
Washington, DC 20231

Sir:

This is in response to the Office Action dated December 30, 1999 in the above-identified patent application. Reconsideration of the above-identified patent application in light of the following remarks is respectfully requested. Claims 33-38 have been rejected under 35 U.S.C. § 112, first paragraph, as "containing subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention." Specifically, the Office Action contends that the recitations in claims 33-38 of "a remote computer having a remote memory for storing a program and a remote revision status, a main computer having a main memory for storing the latest revisions of the program and a main program revision status, etc" are "not fully supported by the specification and/or any related application."

Applicants respectfully submit that the specification of Applicants' above-identified application as well as the U.S. priority applications therefor (the above-identified application is a continuation of the Filepp et al. U.S. priority application Serial No. 08/740,043, which is a division of application Serial No. 08/158,026, all of which have the same disclosure)

fully describe and support all of the limitations of claims 33-38. Support for claims 33-38 in the specification of the Filepp et al. application is shown in detail in the Preliminary Amendment filed September 18, 1997 as well as by the following chart (the left column thereof reciting claim elements and the right column thereof indicating the support in the Filepp et al. priority application therefor):

Claims Presented	Support In Filepp Et Al. Applications
33.A system for automatically updating a program stored in a remote computer, the system comprising:	The Filepp et al. system provides for, inter alia, updating program instructions (portions of a program) and other information (collectively called "objects") stored in a remote computer.
a remote computer including a remote memory for storing a program and a remote revision status,	Filepp et al. disclose a remote computer (each called a reception system ("RS")) with memory for storing programs having revision indicia in a network that features a plurality of such remote computers for displaying information and providing transactional services to users. P. 7, lines 7-14. Users access the network with a RS which is configured as a conventional personal computer enabled with software in conformity with the invention. The RS includes an INTEL based processor, RAM, ROM, and disk memory. P. 8, lines 1-14. Users access the network with their respective RS through a modem. P. 14, lines 5-7. The RS includes means for selectively storing programs and display data in the form of objects. The objects are stored at the RS in accordance with predetermined storage criteria. P. 10, lines 13-19. Filepp et al. disclose that "[o]bjects carry application program instructions and/or information for display at [the] monitor screen... of [the remote] RS." P. 9, lines 29-30. Filepp et al. disclose that the revision indicia of a stored program is maintained with the object containing the program: "to effect object storage management, objects are provided with a coded version id made up of the storage control byte and version control bytes" which are "elements of the object header." P. 135, lines 22-25. The currency of objects stored at the RS is established by virtue of the object's storage control parameters and a check of the object's version identification prior to use. P. 10,lines 13-19. The terms "version" and "revision" are equivalents in the art. Alan et al. *Computer Desktop Encyclopedia*. Both terms denote the "currency" of the data to which they are respectively applied.
the remote program revision status indicating the revision level of the program stored in the remote memory;	Filepp et al. disclose remote program revision indicia stored with the object containing the program, i.e., at the remote computer for objects stored at the remote computer: "to effect object storage management, objects are provided with a coded version id made up of the storage control byte and version control bytes" which are "elements of the object header." P. 135, lines 22-25. The currency of objects stored at the RS is established by virtue of the object's storage control parameters and a check of the object's version identification prior to use.

Claims Presented	Support In Filepp Et Al. Applications
	P. 10,lines 13-19.
a main computer including a main memory for storing the latest revisions of the program and a main program revision status,	Filepp et al. disclose a main computer with memory for storing the most current revision indicia of an object (the main computer includes a file server and concentrator which, together, are called the network delivery system). P. 11, line 31 - p. 12, line 4. Information and programs used in a remote RS either reside at the RS and are updated from the network, or are available on demand from the main computer (i.e., from the network delivery system). P. 13, lines 1-10.
the main program revision status indicating the revision level of the program stored in the main memory;	Filepp et al. disclose that the revision indicia of a stored program is maintained with the object containing the program. According to this storage plan, the version id defines two fields, a first field to identify the object version and a second field to identify the object storage candidacy. P. 135, lines 25-28. As noted above, objects carry their version id with them in their respective headers wherever the object is stored. As noted, the most current version of an object is introduced at the network delivery system. P. 13, lines 1-10. Thus, the latest version level of the object will be at the network delivery system i.e., at the main computer.
means for transmitting the remote program revision status from the remote computer to the main computer;	Filepp et al. disclose that their system includes structure for transmitting version indicia from a remote computer to the network delivery system (or main computer). P. 146, line 27 - p. 148, line 15. As explained at p. 139, lines 23-36, as part of the object request procedure, the RS transmits the object version to ascertain the currency of the object. When a remote RS calls an object for use, the RS first sends a request to the delivery system (i.e., main computer) to verify the currency of the requested object. As part of that request, the RS sends the version and object id for the object. P. 139, lines 23-26.
means for comparing the remote program revision status to the main program revision status;	Filepp et al. disclose structure for comparing version indicia of objects stored at the RS and the network delivery system. Specifically, during version checking, when an object stored at a remote computer (RS) is initially fetched or accessed during a session, a request to the delivery system (i.e., the main computer) is made to verify object currency by specifying the version and id of the object stored at the remote computer. P. 139, lines 23-36. In response, the version id for a referenced object (i.e., the object at the remote RS) is compared by the network delivery system to the object version stored at the network delivery system. If the network delivery system determines that the object version id is current it advises the RS that the object can be used. If the network delivery system determines that the object is not current, a new object (i.e., the current object) is sent.
means for determining which portions of the program stored in a main memory are different from the program stored in the remote memory;	Filepp et al. disclose structure for distinguishing between different versions of parts of programs. Filepp et al. describes individual object version checking. P. 139, lines 23-36. Filepp et al. describe the formulation of applications, programs and display data with objects. P. 12, lines 7-22. Objects may contain other objects and may also provide reference to other objects by name. P. 9, line 22 - p. 10 line 4. Program objects are dynamically invoked from other objects, for example, program objects may be called for execution by means of program call segments, "which specify when a program is to be executed (event), what program to execute (program

Claims Presented	Support In Filepp Et Al. Applications
	pointer), and how programs should run (parameters)." P. 18, line 31 - p. 19, line 11. See also p. 13, lines 16-35, concerning objects being portions of applications; e.g., applications are constructed as groups of objects and distributed on demand to a user's RS.
means for transmitting updated portions of the program stored in the main computer to the remote computer;	Filepp et al. disclose structure for transmitting current versions of programs from the main computer to the RS. Thus, for example, in response to a request for an object version check, the network delivery system will advise the RS "either that the version id of the stored object matches the current value; i.e., the stored object is acceptable, or deliver a current object that will replace the stored object shown to be stale." P. 139, lines 23-36.
means for replacing portions of the program stored in the remote computer with updated portions of the program received from the main computer; and	Filepp et al. disclose structure for replacing an outdated portion of a program stored at the RS with a current version received from the main computer. Where a version checked remotely stored object is found to be stale, the new object delivered by the distribution system will replace the old one. P. 139, lines 27-30.
means for transmitting the new program revision status from the main computer to the remote computer.	Because the version id is a part of the object header and accordingly the object itself (P. 135, lines 22-25 described above), the new version indicia is transmitted from the main computer to the RS when a new object is transmitted.
34. The system of claim 33, further comprising	
means for storing and maintaining variable data and constant data related to a plurality of products in the memory of the main computer,	Filepp et al. disclose a network delivery system (i.e., main computer) which stores data for delivery to a requesting remote RS, and routes data entered by the user or collected at the RS to the network. P. 11, line 31 - p. 12, line 4. Filepp et al. disclose, at p. 137, line 6 - p. 138, line 26, that objects can have different storage candidacy values which dictate whether and for how long objects (program instructions and/or data) are stored at the RS. Filepp et al's first or second values correspond to "variable data" and actually provide for at least two different degrees of variable data: A first candidacy value is applied where the object is very sensitive to time; e.g., news items, volatile pricing information such as might apply to stock quotes, etc. In accordance with this first value, the object will not be permitted to be stored on RS 400, and RS 400 will have to request such objects from delivery system 20 each time it is accessed, thus, assuring currency. A second value is applied where the object is sensitive to time but less so than the first case; e.g., the price of apples in a grocery shopping application. Here, while the price might change from day to day, it is unlikely to change during a session. Accordingly the object will be permitted to persist in RAM or at the disk cache during a session, but will not be permitted to be maintained at RS 400 between sessions. P. 137, lines 8-19. Filepp et al's third or fifth values corresponds to "constant data":

Claims Presented	Support In Filepp Et Al. Applications
	[W]here the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25. Where the object is of a type required to be stored at RS 400, as for example, objects needed to support standard screens, it is coded for storage between sessions ... However, where such objects are likely to change in the future they may be required to be version checked the first time they are accessed in a session and thus [are] given a fifth storage candidacy value. P. 138, lines 1-7. Variable data thus does not persist at the remote RS beyond, at most, a particular user session; it is retrieved from the network delivery system when it is needed. Constant data is stored locally but is version checked when accessed. Thus, the most current constant data is always stored on the network.
means for storing constant data related to a plurality of products in the memory of the remote computer,	An object storage facility provided in the RS software manages objects remotely stored in a local store including a cache (segmented between available RAM and a fixed size disk file) and stage (fixed size disk file). P. 133, line 30 - p. 134, line 28. Objects stored in the stage file are retained between sessions (and thus are "constant data"). The storage control field, located in the header portion of an object, described more fully hereafter as the object "storage candidacy", indicates whether the object is variable or constant: Stageable objects [i.e., constant data] must not be subject to frequent change or update. They are retained between user sessions on the system. . . Cacheable objects [i.e., variable data] can be retained during the current user session, but cannot be retained between sessions. These objects usually have a moderate update frequency. . . Trashable objects [also representing variable data] can be retained only while the user is in the context of the partitioned application in which the object was requested. Trashable objects usually have a very high update frequency and must not be retained to ensure that the user has access to the most current data. . . Specifically, to effect object storage management, objects are provided with a coded version id made up of the storage control byte and version control bytes identified above as elements of the object header. . . P. 134, line 2 - p. 135, line 25. Filepp et al. also disclose that constant data is stored at the remote computer:

Claims Presented	Support In Filepp Et Al. Applications
	[Where] the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25. Where the object is of a type required to be stored at RS 400, as for example, objects needed to support standard screens, it is coded for storage between sessions ... However, where such objects are likely to change in the future they may be required to be version checked the first time they are accessed in a session and thus [are] given a fifth storage candidacy value. P. 138, lines 1-7.
means for selecting a product from the remote computer memory for which product information is desired,	Filepp et al. disclose that a user, through a remote RS, can obtain information and perform transactions regarding a wide variety of products and services. P.10, line 33 - p. 11, line 1. See also p. 137, lines 8-19. Filepp et al. give the example of a user at a remote computer purchasing an apple through the network. At p.137, lines 13-19, the price of the apple is described as data transmitted from the network delivery system or main computer (i.e., variable data) because it changes so frequently that there is no point in storing it locally. At p. 148, line 26 - p. 153, line 10, the entire procedure by which the user interacts with the remote computer and the network to purchase apples is detailed. Again, at p. 149, line 36, the price of an apple is obtained from the network delivery system (or main computer) after being selected at the remote. The presentation data etc. related to the interactive apple purchase (i.e., constant data) is stored remotely because it does not change frequently. The constant presentation data etc. related to the purchase of apples is clearly shown in Filepp Fig. 3b, with blank spaces for the variable price data which is ultimately transmitted from the network computer.
means for comparing constant data in the memory of the remote computer with constant data in the memory of the main computer,	Filepp et al. disclose that information stored at a remote computer is compared with corresponding information available at the network delivery system: [Where] the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25. Filepp et al. disclose structure for comparing version indicia of objects stored at the RS and the network delivery system. Specifically, during version checking, when an object stored at a remote computer (RS) is initially fetched or accessed during a session, a request to the delivery system (i.e., the main

Claims Presented	Support In Filepp Et Al. Applications
	computer) is made to verify object currency by specifying the version and id of the object stored at the remote computer. P. 139, lines 23-36. In response, the version id for a referenced object (i.e., the object at the remote RS) is compared by the network delivery system to the object version stored at the network delivery system. If the network delivery system determines that the object version id is current it advises the RS that the object can be used. If the network delivery system determines that the object is not current, a new object (i.e., the current object) is sent.
means for determining updated portions of the constant data stored in the main computer that are different than the constant data stored in the remote computer,	As noted with respect to claim 33, above, Filepp et al. disclose that their system includes structure for transmitting version indicia from a remote computer to the network delivery system (or main computer). P. 146, line 27 - p. 148, line 15. As explained at p. 139, lines 23-36, as part of the object request procedure, the RS transmits the object version to ascertain the currency of the object. When a remote RS calls an object for use, the RS first sends a request to the delivery system (i.e., main computer) to verify the currency of the requested object. As part of that request, the RS sends the version and object id for the object. P. 139, lines 23-26. Filepp et al. disclose structure for distinguishing between different versions of parts of programs. Filepp et al. describe individual object version checking. P. 139, lines 23-36. Filepp et al. describe the formulation of applications, programs and display data with objects. P. 12, lines 7-22. Objects may contain other objects and may also provide reference to other objects by name. P. 9, line 22 - p. 10 line 4. Program objects are dynamically invoked from other objects, for example, program objects may be called for execution by means of program call segments, "which specify when a program is to be executed (event), what program to execute (program pointer), and how programs should run (parameters)." P. 18, line 31 - p. 19, line 11. See also p. 13, lines 16-35, concerning objects being portions of applications; e.g., applications are constructed as groups of objects and distributed on demand to a user's RS.
means for transmitting the updated portions of the constant data stored in the main computer from, the main computer to the remote computer, and	Filepp et al. disclose structure for comparing version indicia of objects stored at the RS and the main computer. Specifically, during version checking, when an object stored at a remote computer (RS) is initially fetched or accessed during a session, a request to the delivery system (i.e., the main computer) is made to verify object currency by specifying the version and id of the object stored at the remote computer. P. 139, lines 23-36. In response, the version id for a referenced object (i.e., the object at the remote RS) is compared by the network delivery system to the object version stored at the network delivery system. If the network delivery system determines the object version id is current it advises the RS that the object can be used. If the network delivery system determines the object is not current, a new object (i.e., the current object) is sent. As described above, Filepp et al. disclose that the version of constant data available locally at the RS is compared to the version stored at the network delivery system and updated at the RS if necessary. P. 133, lines 7-13. Filepp et al. disclose structure for transmitting current versions of programs from the main computer to the RS. Thus, for

Claims Presented	Support In Filepp Et Al. Applications
	example, in response to a request for an object version check, the network delivery system will advise the RS "either that the version id of the stored object matches the current value; i.e., the stored object is acceptable, or deliver a current object that will replace the stored object shown to be stale." P. 139, lines 23-36. Additionally, Filepp et al. note that the RS includes means for communicating with the network to retrieve objects in response to events occurring at RS. P. 10, lines 10-12.
means for replacing portions of the constant data stored on the remote computer with the updated portions of constant data received from the main computer.	Filepp et al. disclose structure for comparing version indicia of objects stored at the RS and the main computer. Specifically, during version checking, when an object stored at a remote computer (RS) is initially fetched or accessed during a session, a request to the delivery system (i.e., the main computer) is made to verify object currency by specifying the version and id of the object stored at the remote computer. P. 139, lines 23-36. In response, the version id for a referenced object (i.e., the object at the remote RS) is compared by the network delivery system to the object version stored at the network delivery system. If the network delivery system determines the object version id is current it advises the RS that the object can be used. If the network delivery system determines the object is not current, a new object (i.e., the current object) is sent. As described above, Filepp et al. disclose that the version of constant data available locally at the RS is compared to the version stored at the network delivery system and updated at the RS if necessary. P. 133, lines 7-13. Filepp et al. disclose structure for replacing an outdated portion of a program stored at the RS with a current version received from the main computer. Where a version checked remotely stored object is found to be stale, the new object delivered by the distribution system will replace the old one. P. 139, lines 27-30. Filepp et al. disclose structure for transmitting current versions of programs from the main computer to the RS. Thus, for example, in response to a request for an object version check, the network delivery system will advise the RS "either that the version id of the stored object matches the current value; i.e., the stored object is acceptable, or deliver a current object that will replace the stored object shown to be stale." P. 139, lines 23-36. Additionally, Filepp et al. note that the RS includes means for communicating with the network to retrieve objects in response to events occurring at RS. P. 10, lines 10-12.
35. The system of claim 34, further comprising	
means for integrating constant data stored in the memory of the remote computer associated with the selected product with the variable data received from the main computer to provide information related to the selected product including both constant and variable data.	As described above, Filepp et al. give the example of a user at the RS purchasing an apple through the network delivery system. At p. 137, lines 13-19, the price of the apple is variable data which changes so frequently that there is no point in storing it at the RS. At p. 148, line 26 - p. 153, line 10, the price of an apple is obtained from the network delivery system (or main computer) after being selected from the RS. The presentation data etc. related to the interactive apple purchase is constant data and is stored remotely because it does not change frequently. The constant presentation data etc. related to the purchase of apples is clearly shown in Filepp Fig. 3b, with blank spaces for the variable price data transmitted from

Claims Presented	Support In Filepp Et Al. Applications
	the network delivery system. Thus, the constant data related to an apple purchase is integrated with variable data related to, e.g., the price of an apple obtained from the network delivery system.
36.The system of claim 34, further comprising	
means for storing and maintaining a main constant data revision status in the memory of the main computer,	As noted above, objects carry their version id with them in their respective headers and accordingly, wherever the object is stored. Thus, the revision status of constant data is stored at the main computer with each object. Filepp et al's objects include, inter alia, program instructions (portions of a program) and data. P. 8, lines 25-28; p. 9, lines 29-30. As noted above, the main computer has a memory for storing the most current version indicia of an object. P. 11, line 31.- p. 12, line 4. Filepp et al. disclose that objects assigned specific storage candidacy values are stored at the remote computer and updated from the main computer as necessary: [Where] the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25.
the main constant data revision status indicating the revision level of the constant data stored in the main computer, and	The revision indicia of a stored program is maintained with the object containing the program. To effect object storage, objects are provided with a coded version id made up of the storage control byte and version control bytes identified above as elements of the object header. P. 135, lines 22-28. As noted, the most current version of an object is introduced at the network delivery system. P. 13, lines 1-10. Since the object's version id is part of the object, the latest version level of the object will be at the network delivery system. Filepp et al. disclose that objects assigned specific storage candidacy values are stored at the remote computer and updated from the main computer as necessary: [Where] the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25.
means for storing a remote constant data revision status in the memory of the remote computer,	As noted with respect to claim 33, above, The RS includes means for selectively storing program instructions and display data in the form of objects. The objects are stored at the RS in accordance with a predetermined storage criteria. P. 10, lines 13-19. Filepp et al. disclose that "[o]bjects carry application program instructions and/or information for display at [the] monitor screen... of [the remote] RS." P. 9, lines 29-30. Filepp et al. disclose that the revision indicia of a stored

Claims Presented	Support In Filepp Et Al. Applications
	program is maintained with the object containing the program: "to effect object storage management, objects are provided with a coded version id made up of the storage control byte and version control bytes" which are "elements of the object header." P. 135, lines 22-25. The currency of objects stored at the RS is established by virtue of the object's storage control parameters and a check of the objects version identification prior to use. P. 10,lines 13-19 As above, since a data object's version id is part of the object, the version for the object is stored wherever the object is stored. For constant data stored at the remote computer, revision status is also stored there.
the remote constant data revision status indicating the revision level of the constant data stored in the remote computer.	Filepp et al. disclose that objects assigned specific storage candidacy values are stored at the remote computer: [Where] the object concerns information sufficiently stable to be maintained between sessions, a third storage candidacy value is set to permit the object to be stored at RS 400 between sessions, on condition that the object will be version check[ed] the first time it is accessed in a subsequent session. P. 137, lines 20-25.
37. The system of claim 36, wherein the means for comparing constant data in the memory of the remote computer with constant data in the memory of the main computer compares the remote constant data revision status with the main constant data revision status maintained in the main computer.	As described above, Filepp et al. disclose that the version of objects available locally at the RS is compared to the version stored at the network delivery system and updated at the RS if necessary. P. 133, lines 7-13.
38.The system of claim 37, wherein the means for transmitting updated portions of the constant data stored in the main memory from the main computer to the remote computer also transmits an updated remote revision status identical to the main revision status from the main computer to the remote computer.	Because the version id is a part of the object header and accordingly the object itself (P. 135, lines 22-25 described above), the new version indicia is transmitted from the main computer to the remote RS when a new object is transmitted.

As detailed above, claims 33-38 are amply supported by the disclosure. No other rejections being outstanding, the application is thus in condition for allowance. Reconsideration of the application and allowance of claims 33-38 is therefore respectfully requested. Should any questions remain, the Examiner is requested to contact the undersigned attorney for Applicants to discuss them, if it is believed such a discussion would assist the Examiner in assessing Applicants' claim of support for claims 33-38.

Respectfully submitted,

March 30, 2000

Israel Blum
Registration No. 26,710

MORGAN & FINNEGAN, L.L.P.
345 Park Avenue
New York, New York 10154
(212) 758-4800
(212) 751-6849 (FAX)